

Chapter 16

PORTFOLIO PLANNING

Most organizations want or need to produce more than one product at a time. These multiproduct organizations need a way to make economically sound choices regarding how to manage their product portfolios. They also need their portfolio management or governance processes to align well with core agile practices; otherwise, there will be a fundamental disconnect with the agile approach being used at the individual product level. This chapter lays out 11 strategies for portfolio planning, grouped by scheduling, product inflow, and product outflow. It ends with a discussion of how to determine whether or not more work should be invested in in-process products.

Overview

Portfolio planning (or portfolio management) is an activity for determining which portfolio backlog items to work on, in which order, and for how long. A portfolio backlog item can be a product, a product increment (one release of a product), or a project (if your organization prefers to plan work around projects). In this chapter I use the word *product* generically to mean all types of portfolio backlog items.

In my experience, most organizations (agile or otherwise) do a very poor job of portfolio-level planning. Many have portfolio-level planning processes that are fundamentally at odds with core agile principles. When this happens, decisions are made at the portfolio level that disrupt the fast, flexible flow of work. In this chapter I discuss how to avoid this disconnect by performing portfolio planning in a manner that is well aligned with core agile principles.

Timing

Portfolio planning is a never-ending activity. As long as we have products to develop or maintain, we have a portfolio to manage.

As shown in Figure 15.1, portfolio planning deals with a collection of products and is therefore larger in scope and higher level than individual product-level planning (envisioning). Being higher level, however, doesn't mean that portfolio planning precedes product planning. In fact, the output of planning or envisioning a new product is an important input to portfolio planning. Using data from envisioning, portfolio planning determines whether to fund the product and how to sequence it into the **portfolio backlog**. Portfolio planning isn't just for newly envisioned products, though. It also occurs at regularly scheduled intervals to review products that are already in process (under development, already live in production, or currently being sold).

Participants

Because portfolio planning focuses on both new products and in-process products, its participants include an appropriate set of internal stakeholders, the product owners of individual products, and optionally, but frequently, senior architects and technical leads.

The stakeholders must have a sufficiently broad business perspective to properly prioritize the portfolio backlog and make decisions regarding in-process products. In some organizations the stakeholders collectively form an approval committee, governance board, or some equivalent entity that oversees the portfolio-planning process.

Product owners also participate in portfolio planning as champions of their respective products and advocates for necessary resources.

Frequently the input of senior architects or technical leads is needed to ensure that important technical constraints are factored into portfolio-planning decisions.

Process

Figure 16.1 illustrates the portfolio-planning activity.

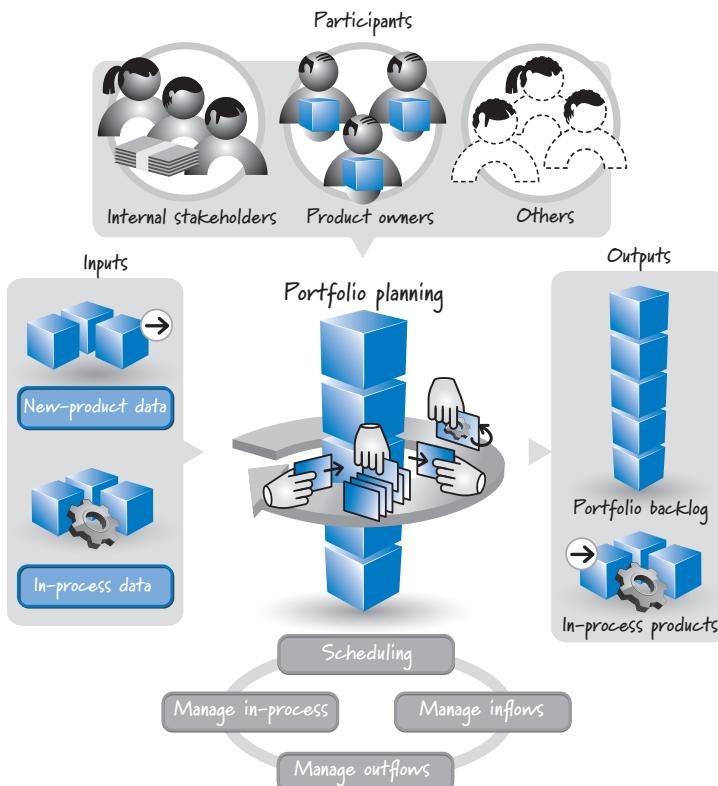


FIGURE 16.1 Portfolio-planning activity

As I stated earlier, inputs to portfolio planning include newly envisioned products (candidates for inclusion in the portfolio backlog) and in-process products. The new products come with data that was gathered during envisioning, such as cost, duration, value, risk, and so on. In-process products come with their own set of data, such as intermediate customer feedback, updated cost, schedule, and scope estimates, technical debt levels, and market-related data, which will help determine the path forward for these products.

Portfolio planning has two outputs. The first is the portfolio backlog, which is a prioritized list of future products, ones that have been approved but for which development has not yet begun. The second is a set of active products—new products that have been approved and are slated for immediate development, as well as products that are currently in process and have been approved to continue.

To arrive at these outputs, participants engage in four categories of activities: scheduling, managing inflows, managing outflows, and managing in-process products. Figure 16.2 summarizes the specific strategies associated with each of these categories.

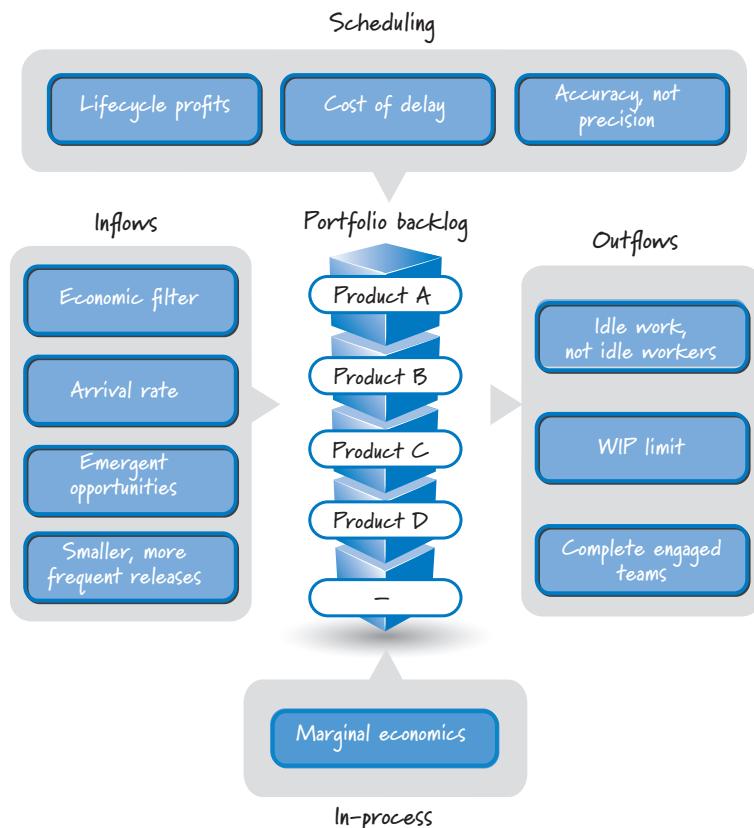


FIGURE 16.2 Portfolio-planning strategies

The scheduling strategies help determine the proper sequence of the products in the portfolio backlog. Inflow strategies guide participants in knowing when to insert items into the portfolio backlog. Outflow strategies inform participants about when to pull a product out of the portfolio backlog. The in-process-product strategy is used to decide when to preserve, pivot, deliver, or terminate a product that is currently in process.

The remainder of this chapter will discuss the 11 strategies that make up these four categories.

Scheduling Strategies

Portfolio planning must allocate an organization's limited amount of resources to products in an economically sensible way. Although there are many ways to decide the sequence of products, I focus on three strategies:

- Optimize for lifecycle profits.
- Calculate the cost of delay.
- Estimate for accuracy, not precision.

Optimize for Lifecycle Profits

To optimize product ordering within the portfolio we need to decide which variable to measure so that we can determine whether our optimization efforts are working. Reinertsen recommends that we use an economic framework where we consider all decisions and trade-offs in a standardized and useful unit of measure: **lifecycle profits** (Reinertsen 2009b). Based on this recommendation, our goal should be to sequence the items in the portfolio backlog to maximize overall lifecycle profits.

For a specific product, lifecycle profits are the total profit potential for the product over its lifetime. In the case of portfolio planning, we are interested in optimizing the lifecycle profits of the entire portfolio rather than a single product. Thus, we might have to suboptimize individual products in order to optimize the portfolio (Poppendieck and Poppendieck 2003). So the goal of the strategy of optimizing for lifecycle profits is to find the sequence of portfolio backlog items that provides the greatest lifecycle profits across the entire portfolio (see the next section on calculating the cost of delay for an example).

Reinertsen further asserts that the two variables most important to assessing the impact on lifecycle profits are cost of delay and duration (a good common proxy of which is effort or product size). Based on how similar these variables are (or aren't) across products in the portfolio, he suggests selecting one of the three scheduling approaches shown in Table 16.1.

When all products have the same cost of delay, the preferred scheduling strategy is to do the shortest job first. When products are the same size (have the same

TABLE 16.1 Different Portfolio Scheduling Principles

(If) Cost of Delay	(And) Duration/Size	(Then) Scheduling Approach
Same across all products	Varies across products	Shortest job first
Varies across products	Same across all products	High delay cost first
Varies across products	Varies across products	Weighted shortest job first

duration), the preferred scheduling strategy is to first work on the products with a high cost of delay. When both cost of delay and duration can vary (which is the normal case in product development), the economically optimal sequencing is achieved using **weighted shortest job first (WSJF)**, which is calculated as cost of delay divided by duration (or effort to implement).

Next I will discuss both cost of delay and estimating the effort/cost of products in the portfolio.

Calculate Cost of Delay

When we sequence items in the portfolio backlog, we must necessarily work on some products before we work on others. Those that we don't work on immediately have a delayed start and therefore a delayed delivery date, for which there exists a quantifiable cost.

As I described in Chapter 3, the cost of delay provides essential information for making informed economic decisions. Yet most organizations aren't even in a position to answer a question as simple as "If we delay the product deployment by one month, what would be the cost of that delay in lifecycle profits?"

Being blind to the cost of delay, most organizations choose to sequence their portfolio using the simple (and frequently wrong) approach of "high profit first" (see Table 16.2).

In this example, project A has a 20% ROI and project B has a 15% ROI. Using the high-profit-first scheduling strategy, we would do project A before project B because it has the higher return on investment. Although this approach seems sensible, it fails to take into account the cost of delay of each product, which could substantially alter

TABLE 16.2 Example of Using Cost of Delay to Sequence the Portfolio

	Project A	Project B
Return on investment (ROI)	20%	15%
Cost of delay (1 month)	\$5K	\$75K

the lifecycle profit calculation. For example, what if project A has a \$5K/month cost of delay and project B has a \$75K/month cost of delay (as shown in Table 16.2)? In this case, delaying project B to work first on project A has a much greater impact on portfolio lifecycle profitability.

Cost of delay embodies the fact that time does or can affect most variables. In the previous example, the ROIs of project A and project B were computed using specific, time-dependent assumptions (for example, when development would start and end, what resources would be available at that time, how much the resources would costs, what prices people would be willing to pay to purchase the product over time, what technology and business risks would exist and what probabilities of occurrence and cost impacts they would have). Delay or accelerate development and the values of these variables can and frequently do change. So, cost of delay is not the only factor to consider when prioritizing items in the portfolio; instead, it is the time dimension that must be considered because it affects all other prioritization variables such as cost, benefit, knowledge, and risk.

The most frequent complaint I hear about cost of delay is that it is not clear how it should be calculated. Most of the time this concern is unfounded because running two different spreadsheet models that calculate profitability (one without a delay and one with a delay) should effectively calculate the cost of delay.

Leffingwell offers a model for calculating cost of delay that is the aggregation of three product attributes (Leffingwell 2011):

- User value—potential value in the eyes of the user
- Time value—how user value decays over time
- Risk reduction/opportunity enablement—the value in terms of mitigating a risk or exploiting an opportunity

To calculate the cost of a delay for a product, each of these three attributes is assigned its own individual cost-of-delay number using a scale of 1 (lowest) to 10 (highest). The total product cost of delay is the sum of the three individual delay costs.

An alternative, and frequently effective, approach for making informed scheduling decisions is to characterize the general profile of the delay cost (see Figure 16.3).

Table 16.3 describes each of these profiles in more detail.

If calculating a precise cost-of-delay number is very time-consuming or error-prone, consider selecting an appropriate delay profile (or creating a new one) and use that profile instead of a specific delay number when making scheduling decisions.

Does cost of delay apply in organizations that develop products in highly regulated industries like medical devices or health care where compliance and patient safety are critical? Such critical factors must be considered when determining product priorities; however, important properties of these factors can be accounted for using cost of delay expressed in terms of lifecycle profits.

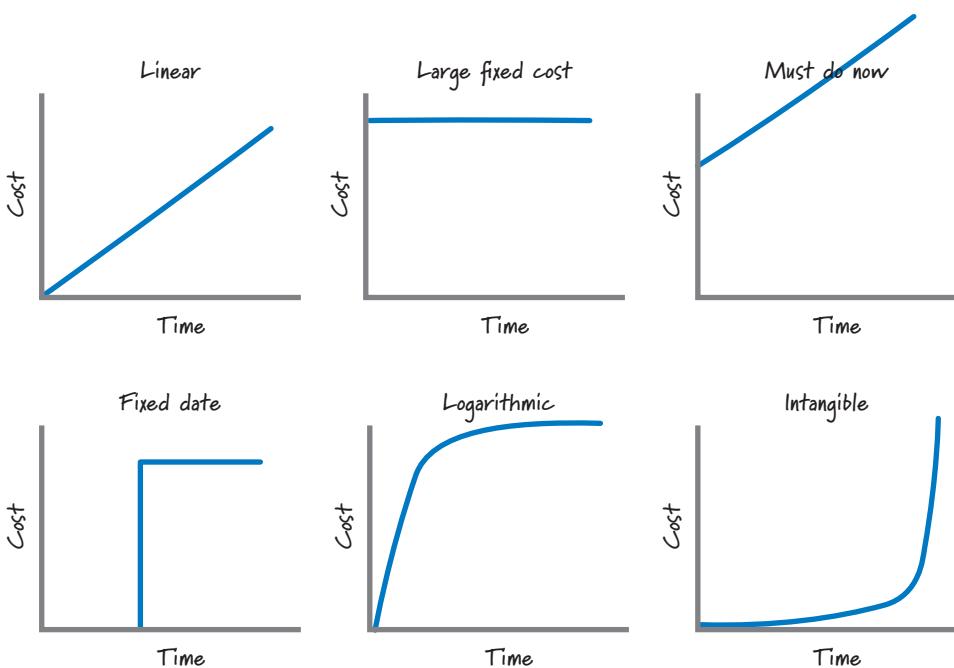


FIGURE 16.3 Cost-of-delay profiles

TABLE 16.3 Description of Cost-of-Delay Profiles

Profile Name	Description
Linear	A product with a cost of delay that increases at a constant rate.
Large fixed cost	A product that accrues a one-time cost if not acted on immediately; for example, we receive a substantial payment only after the product is delivered.
Must do now	A product that we “must do now” because we experience an immediate and aggressively increasing cost of delay; for example, a product without which we incur immediate lost revenue or cost savings that continue to grow over time.
Fixed date	A product that must be delivered by a fixed future date and therefore has a zero cost of delay until the fixed date occurs. After the fixed date passes, the full cost of at least the initial delay is accrued.
Logarithmic	A product that accrues most of the delay cost very early, with increasingly less incremental delay thereafter.

continues

TABLE 16.3 Description of Cost-of-Delay Profiles (*Continued*)

Profile Name	Description
Intangible	A product (or body of work) that has no “apparent” cost of delay for an extended period of time and then, suddenly, accrues a very high delay cost. An example would be how many organizations treat technical debt. Today there appears to be little or no cost of delay of not repaying the technical debt. However, as I described in Chapter 8, technical debt can reach a tipping point, at which time the cost of delaying other work is noticeable and very high.

For example, in the United States, organizations such as health plans and health-care providers must use certain codes to identify specific diagnoses and clinical procedures on claims, encounter forms, and other electronic transactions. The standard for these codes at the time this book was written is *International Classification of Diseases*, 9th revision (ICD-9-CM). However, a new standard, ICD-10-CM, will replace ICD-9-CM on October 1, 2013. At that time, organizations that are subject to U.S. HIPPA (Health Insurance Portability and Accountability Act of 1996) regulations must comply with ICD-10-CM. Many such organizations have a portfolio of products that need remediation—in a fashion strikingly similar to the Year 2000 (Y2K) problem at the turn of the millennium. Because all products in the remediation portfolio have a fixed-date cost-of-delay profile (as shown in Figure 16.3), to rationally sequence the remediation work, these organizations need to consider what the cost of the delay (in lifecycle profits) would be for each product if the remediation work on that product is *not* completed by October 1, 2013. For example, a critical product that is not in compliance could generate a loss of \$100M/year, whereas another noncompliant product might generate a loss of \$5M/year. So calculating cost of delay is a critical variable for sequencing the remediation portfolio in an economically sensible way.

Estimate for Accuracy, Not Precision

To properly schedule portfolio backlog items we also need to understand their effort/cost (because cost affects lifecycle profits). When estimating the size of portfolio backlog items, we are looking for accuracy, not precision, because of the very limited data we have at the time when a first estimate is required.

In Chapter 7 I discussed the fact that some organizations prefer to estimate portfolio backlog items using T-shirt sizes instead of overly precise numbers. Each T-shirt size corresponds to an associated cost range (see Table 16.4 for an example of one organization’s mapping).

In this table, the rough cost range includes the labor cost (which typically represents the majority of a product’s cost in this organization) as well as capital expenditures and any other costs deemed material to the product development effort.

TABLE 16.4 Example of T-Shirt Size Estimation

Size	Rough Cost Range
Extra-small (XS)	\$10K to \$25K
Small (S)	\$25K to \$50K
Medium (M)	\$50K to \$125K
Large (L)	\$125K to \$350K
Extra-large (XL)	>\$350K

The benefit of T-shirt size estimation is that it's fast, usually accurate enough, and provides actionable information at a portfolio level.

How accurate is accurate enough? Let me give you an example. In the aforementioned organization, the engineering department had spent considerable time in the past trying to give very precise estimates. People weren't sure whether T-shirt sizes would be accurate enough, but everyone agreed to give them a try. Soon after, marketing came to engineering with an idea for a project; the engineering department discussed it and assigned it a size, Medium.

Marketing was then able to decide if the benefit of doing the project exceeded the cost of a medium-size project (\$50K to \$125K). This was just as helpful as it had been back when engineering spent a great deal of time to come up with a more precise-sounding, but inaccurate, guess of \$72,381.27. This organization found that the ranges were accurate enough and eliminated waste, without raising expectations too high or providing a false sense of security.

Inflow Strategies

As I will discuss in Chapter 17, the envisioning process details the vision for a new product and collects a set of information that the decision makers need in order to make a go/no-go funding decision. Inflow strategies deal with how to apply the organization's economic filter to make a go/no-go decision. They also deal with how to balance the rate at which products are inserted into the portfolio backlog against the rate at which they are being pulled out, how to quickly embrace an emergent opportunity when it appears, and how to prevent portfolio bottlenecks by using smaller, more frequent releases.

Apply the Economic Filter

The output from envisioning is a product vision along with the information needed to clear the confidence threshold associated with the envisioning (product-planning)

activity (see Chapter 17). This output is the new-product data that is an input to portfolio planning (see Figure 16.1). Based on this data, the organization needs to make a go/no-go decision for moving forward with development of the product. I refer to this activity as applying the economic filter to the new product to see if it meets the organization's funding requirements (see Figure 16.4).

Although each organization needs to define an economic filter that best matches its particular funding policies, a good economic filter should quickly indicate approval of any opportunity that delivers overwhelming value relative to its cost; most everything else (unless there are extenuating circumstances) should be rejected. If the resulting value of developing the product completely overwhelms the costs of developing it, it shouldn't be necessary to spend any significant time discussing it—just approve it and move on to sequencing it into the portfolio backlog. If we find ourselves squabbling over a small difference in cost or value before we can make a decision, we should reject the product because there is clearly not overwhelming economic support for developing it. In most organizations there are simply too many high-value product development opportunities to waste time discussing questionable opportunities.

Balance the Arrival Rate with the Departure Rate

In practice we want a steady stream of products moving into the portfolio backlog balanced against a steady stream of products being pulled from the portfolio backlog (see Figure 16.5).

What we don't want is to overload the portfolio backlog by inserting too many products into it at the same time. This has the effect of overwhelming the system.

To illustrate why, say you want to go to dinner at your favorite restaurant. You get in your car and you drive over. When you arrive, you notice that a large tour bus full of hungry senior citizens has just unloaded and gone into the restaurant.

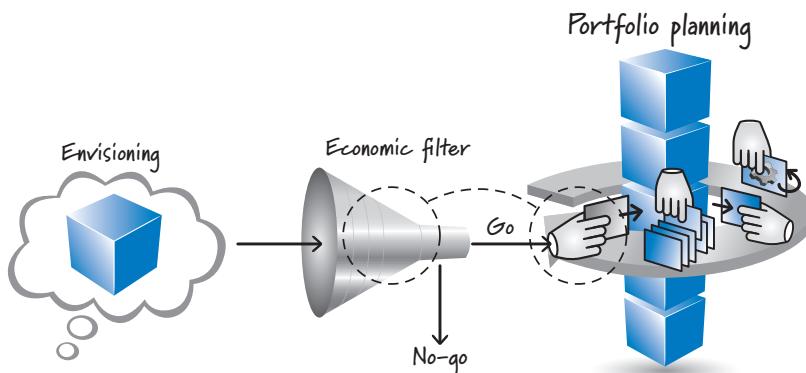


FIGURE 16.4 Applying the economic filter

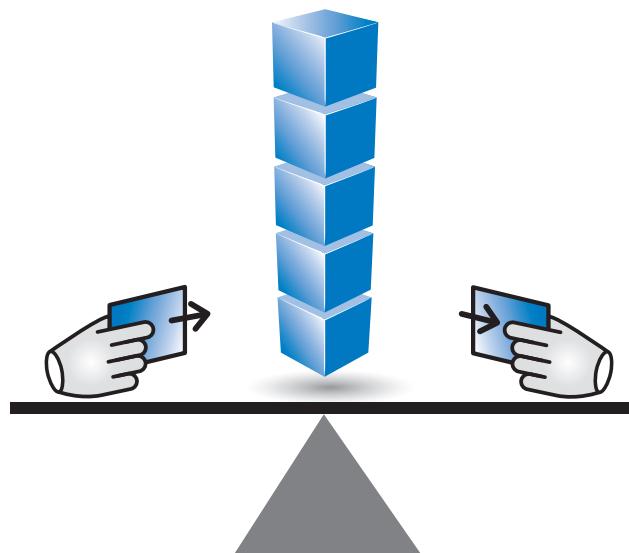


FIGURE 16.5 Balancing inflow and outflow in the portfolio backlog

What would you do? Are you going to enter the restaurant and attempt to dine there? If so, what do you think the consequences of all those hungry seniors descending on the restaurant at the same time will be? Chances are they will overwhelm the restaurant's capabilities. If you risk dining there, you will likely suffer a long and rather unsatisfying experience. Perhaps you should get back in your car and go to another restaurant!

Many organizations conduct an annual strategic-planning event, usually sometime during the third quarter of their fiscal year. Frequently one of the outcomes of strategic planning is a complete list of the products that the organization will work on during the next fiscal year. These products then get simultaneously inserted into the portfolio backlog, typically overwhelming the portfolio-planning process.

I am not suggesting that organizations shouldn't do strategic planning. They should define their strategic direction, but just not *all* of the specific details (at a product level) of how they will achieve that strategy. Deciding at a once-a-year meeting everything to work on over the next fiscal year or longer, and then inserting all of those items into the portfolio backlog at the same time, is a critical decision (and in some organizations an irreversible decision) made in the presence of great uncertainty and violates the principle of keeping planning options open until the last responsible moment (see Chapter 14).

Deciding the entire portfolio of products at one time also violates the principle of using economically sensible batch sizes (as discussed in Chapter 3). Processing a large batch of products to determine how to sequence them into the portfolio backlog

is very expensive and potentially wasteful (because we are planning up to a year or more in advance). It is expensive not only because there are a lot of products to process, but also because a large number of items in the portfolio backlog complicate the scheduling I discussed earlier in this chapter. Determining a good sequencing is a much simpler problem when there are fewer items to sequence. In fact, in the presence of just a small number of portfolio backlog items, any sequencing that avoids an overtly dumb prioritization is typically good enough.

To combat overwhelming the portfolio backlog all at once, we can introduce products to the portfolio at more frequent intervals, for example, monthly (or at least quarterly) instead of annually. Doing so significantly reduces the effort (and cost) required to review and insert new products into the portfolio and provides more overall stability and predictability to portfolio planning.

We should also focus on smaller products (see the strategy on smaller, more frequent releases). This should result in a constant stream of products that are completed, thus freeing up capacity to pull new products from the portfolio backlog at a regular pace. This frequent pulling of products from the portfolio backlog will assist in balancing inflow with outflow.

Last, when the size of the portfolio backlog starts to increase, we can start to throttle the flow of products into the portfolio backlog. We can accomplish this by tweaking the economic filter to raise the product approval criteria so that only higher-value products are allowed to pass through the filter. This will reduce the insertion rate to help establish a better equilibrium with the departure rate.

Quickly Embrace Emergent Opportunities

Portfolio planning needs to embrace **emergent opportunities**. An emergent opportunity is one that was previously unknown, or was deemed sufficiently unlikely to occur and therefore not something worth spending money on today.

For example, one organization I worked with participates in the online betting marketplace. Its business is highly regulated by the jurisdictions in which it is permitted to offer its betting exchanges. Regulators around the world are somewhat unpredictable—especially when it comes to gambling—which makes it difficult to know if and when it will ever be possible to offer a product in a particular jurisdiction. Working in this environment, you need to be prepared for emergent opportunities because regulations can change along with a change in a country's governing party.

One such opportunity was the ability to provide an online betting exchange for horse racing in California. California has a significant number of racetracks, making it a very lucrative opportunity should its regulations change (which, by the way, they did—though illegal for years, online exchange betting is legal as of May 2012). If the organization were in the habit of doing strategic planning only once a year in October (before the laws changed), it would have missed the opportunity to exploit this emergent opportunity—unless it was willing to take on the risk of building an exchange for a marketplace that didn't exist and might never materialize.

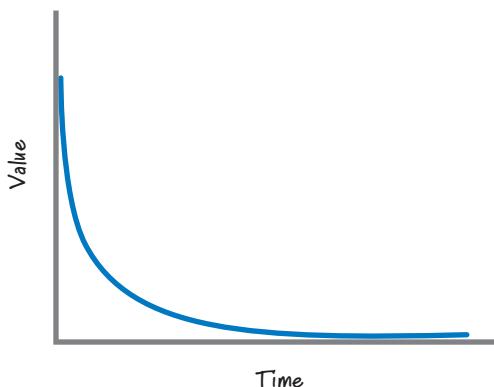


FIGURE 16.6 The value of many emergent opportunities decays rapidly.

Such an emergent opportunity needs to be exploited quickly. Being second to market with an online betting exchange in California would garner little to no market share. Figure 16.6 illustrates this common case where the economic value of an emergent opportunity decays rapidly over time.

By not acting swiftly, as soon as the opportunity becomes available, we immediately lose almost all of the economic value, making it a bad economic choice to pursue it sometime later (such as at the next annual strategic-planning session).

If an organization uses a regular and frequent schedule for evaluating opportunities, such as the once-a-month schedule, and has an efficient economic filter, while at the same time using smaller releases and a WIP limit, it will never have to wait long to consider an emergent opportunity.

Plan for Smaller, More Frequent Releases

As I discussed in Chapter 14, the economics of smaller, more frequent releases are compelling. Figures 14.4 and 14.5, along with Table 14.1, illustrate that we can almost always increase the lifecycle profits of a product if we split a product into a series of smaller, incremental releases.

In addition to this significant benefit, there is another reason we want to manage our portfolio with smaller, more frequent releases—to avoid a convoy effect (see Figure 16.7).

What happens if you are driving on a one-lane country road and you get trapped behind a large farm vehicle (like the one shown in Figure 16.7)? Chances are you and a convoy of smaller vehicles will be trapped behind the larger, slower vehicle for a long time. The cause of the convoy is obvious; the big farm vehicle is hogging the road (the shared resource).

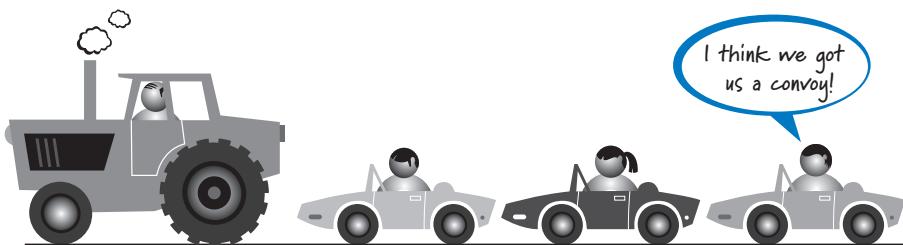


FIGURE 16.7 Large products in the portfolio backlog create a convoy.

This same scenario will occur if we allow large products into the portfolio backlog. Large products require a lot of resources for a considerable amount of time. Those resources are now unavailable to many other smaller products that are caught in the queue behind the large product. And, while caught in the queue, each accrues a cost for being delayed. When we add up the delay costs of all of those small products and then factor in the compelling economics of doing smaller, incremental releases, it becomes clear that large products cause significant economic damage to lifecycle profits.

To combat this issue, some organizations institute a size policy during portfolio planning that specifically limits how large a product development effort may be. One example I encountered was that no product development effort could be larger than nine months. If a proposal was made for a larger product effort, it was summarily rejected and the advocates were told to come up with a way of delivering the product in smaller, more frequent releases.

I have also worked with organizations whose culture is “We can never assume there will be a second release of any product.” This belief is completely at odds with doing smaller, more frequent releases. If we believe that we might never have a second release, the natural reaction is to load up the first release with everything we need, plus everything we think that one day we might need. In this case, not only do we generate larger product development efforts, but we are almost certainly delaying the high-value features of other products while working on the very low-value features of the larger product. This approach is economically damaging. Organizations need to make it clear that follow-on releases can and will be done based on their individual economic merit, and that planning under the assumption of a single release is highly discouraged.

Outflow Strategies

Strategies for managing outflow help organizations decide when to pull a product out of the portfolio backlog. I describe three strategies:

- Focus on idle work, not idle workers.
- Establish a WIP limit.
- Wait for a complete team.

Focus on Idle Work, Not Idle Workers

A key strategy for determining when to pull a product from the portfolio backlog is to remember the principle I discussed in Chapter 3—focus on idle work, not idle workers. This principle states that idle work is far more wasteful and economically damaging than idle workers. This is contrary to how many organizations manage their portfolio.

A common, but misguided, approach to releasing products for development is

1. Pull the top product from the portfolio backlog and assign people to work on it.
2. Are all the people 100% utilized (working at 100% capacity)? If not, repeat step 1.

This approach will keep everyone very busy. What it will also do is keep the work on every product slow and error-prone. A better strategy is to start working on a product only when we can ensure two things: a good flow of work on the new product and that the new product won't disrupt the flow on other in-process products. This strategy is used in close coordination with the next strategy: establish a WIP limit.

Establish a WIP Limit

Consider this scenario. Have you ever gone to a restaurant and seen available tables, yet the staff won't seat you? If you have, you know it's frustrating. Perhaps you think, "Why won't they seat me? They have available tables. Don't they want my business?"

Let's say that several waiters called in sick that day. In that case, a smart restaurateur shouldn't seat you. What happens if he does? Perhaps you have to wait 45 minutes before a server comes to your table. I don't know about you, but I would not be a happy patron if I had to sit for 45 minutes before someone came over to talk to me! I'd actually prefer that they tell me up front, "Sorry, sir, but four of our waiters called in sick today, so it will be 45 minutes before we can seat you." At least this information would give me the option of waiting or going somewhere else.

What would be worse is if they actually seated my party at an available table and then attempted to give us service. If they did that, the service for everyone else in the restaurant would suffer. Seating too many parties relative to the available server capacity would mean that all of the servers will be overworked and everyone will have a bad experience. That's why a smart restaurateur won't seat parties beyond his capacity.

If only we would follow the lead of the smart restaurateur during portfolio planning. We should never pull more products out of the portfolio backlog than we have capacity to complete. Doing so will cause reduced capacity to be available to each product (resulting in each being delayed), as well as cause the quality of work on all products to suffer. Getting work done slower and at lower quality is not a winning strategy.

So how do we determine the appropriate WIP limit? In Chapter 11, I discussed the idea that teams are the unit of capacity that we should use for establishing a WIP limit. Knowing how many Scrum teams and knowing what kinds of products they are capable of working on will guide us as to how many and which types of product development efforts we should pursue simultaneously (see Figure 16.8).

The left side of Figure 16.8 shows that we have three teams that are capable of working on type I products and two teams that can work on type II products. This information would be an excellent starting point for establishing the maximum number of each type of product that our organization can work on simultaneously. Imagine how much more difficult it would be to try to determine the proper number of concurrent development efforts using just the information regarding number of people with particular skill sets (right side of Figure 16.8).

Wait for a Complete Team

The final outflow strategy is to wait for a complete Scrum team to be available before starting to work on a product. Organizations that violate the “Focus on idle work, not idle workers” principle frequently start working on a product when only a couple of people are available. Their thinking might go like this: “Well, a couple of developers aren’t at 100% capacity yet, so let’s have them at least start making progress on that next product.”

This is a flawed strategy because it will cause even more work to get blocked on other products, slowing down all product delivery and generating significant delay costs.

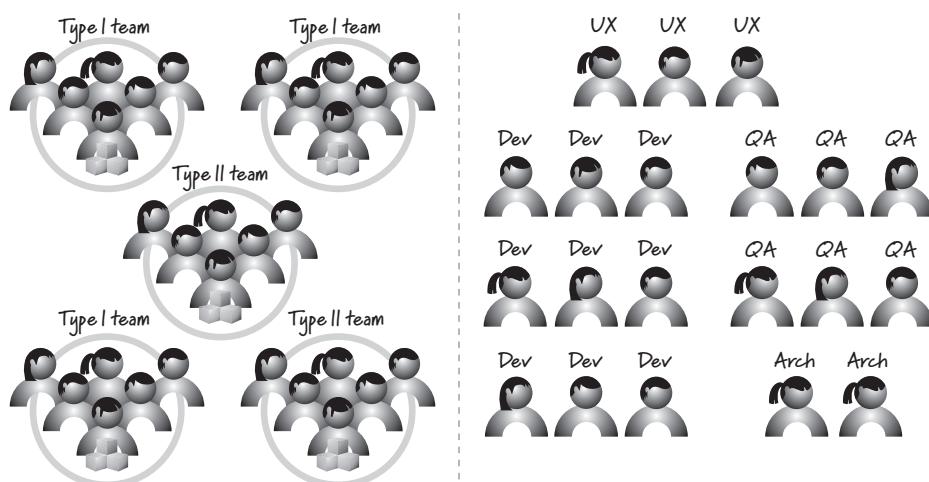


FIGURE 16.8 Teams are the unit of capacity for establishing the product WIP limit.

Because the unit of capacity in Scrum is the team, we shouldn't start working on a product if we don't have a complete Scrum team. Doing so makes no sense from a Scrum perspective. An incomplete Scrum team is insufficient for getting features to a done state.

One variation I would consider is on products that require multiple Scrum teams. Let's say we have a product that requires three Scrum teams. If one complete Scrum team is available and it makes sense to start development with just that one complete team, I would consider starting the product. I would then roll the other full Scrum teams on as they became available.

In-Process Strategies

Strategies for managing in-process products guide us as to whether it is appropriate to preserve, pivot, deliver, or terminate a product that is currently being worked on. We need to make these decisions at regular intervals (say, the end of each sprint) and occasionally at off-cycle times, when abnormal events occur that require us to revisit our in-process products.

There are many different strategies we could consider here, and the governance function of each organization is sure to have its own set of guidelines for dealing with in-process products. However, I will focus on just one strategy—marginal economics. This should be the overarching strategy that guides decision making, and it aligns well with the core Scrum and agile principles I describe in this book.

Use Marginal Economics

From an economic perspective, all work that has been performed on the product up to the decision point is a “sunk cost.” We are interested only in the marginal economics of taking the next step. We should ask ourselves only if spending the next chunk of money is justified by the return that investment would generate. The hard part is making that decision without burdening ourselves with the money we have already spent.

Using marginal economics, we can decide what to do with products that are currently being developed. For each product we scrutinize under the lens of marginal economics, there are four principal options:

- Preserve—continue developing the product.
- Deliver—stop working on the product and ship it.
- Pivot—take what we have learned and change directions.
- Terminate—stop working on the product and kill it.

Figure 16.9 illustrates the decision flow associated with these four options.

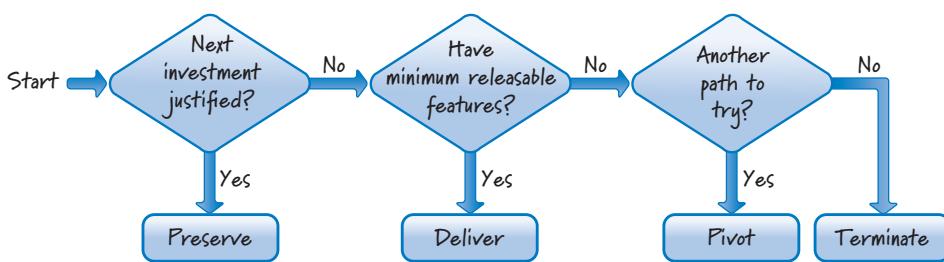


FIGURE 16.9 In-process product decision flow based on marginal economics

If the next investment in the current product is economically justified, *preserving* would be a likely choice. This is the scenario where we review an in-process product and conclude that we should continue spending money on its development.

If further investment in a product is not economically justified, we should decide whether to deliver, pivot, or terminate the product.

If the product we have created thus far contains the minimum releasable features (MRFs), we can consider *delivering* the product. If not, and we are going down the wrong path, and we think there is another path worth exploring, we could *pivot* and change to a new product path. This option would likely involve a return to envisioning to consider the new path (see Chapter 17).

And, if further investment is not justified and we are unhappy with where we are and our prospects for a successful pivot, *terminating* the product would be a viable option.

Foolish behavior can result when marginal economics are ignored. Here's a question to consider: "In your organization, if you spend the first dollar on developing a product, is there any circumstance under which you would terminate development?" I am surprised by the number of times people tell me that their organizations won't ever (or only very rarely) kill a product once the first dollar is spent—in for a penny, in for a pound seems to be their strategy.

At one organization, I was surprised at the explanation for why the company doesn't terminate products. I asked the IT executives, "Suppose you start working on a product that you believe is valuable to 100% of your customers and will cost \$1M to develop. After you have spent \$1M developing it, you learn it would be valuable to only 10% of your customers and will cost a total of \$10M to develop. Would you spend the additional \$9M to complete the product?" Their response: "Yes, we would." My response: "That makes no sense to me! The cost/benefit ratio of this product has changed by a factor of 100. Why would you do that?" Their response: "You don't understand how we do accounting. If we kill the product after we spend \$1M and before the system goes into production, the IT department will suffer a \$1M hit against its expense budget. If we spend the other \$9M and put the system into

production for one day, the full cost of the system moves to the business unit where it can capitalize the expenditure.”

Clearly in this example, gaming the accounting system has trumped common sense.

Marginal economics is a powerful tool for doing the right thing and for exposing foolish and wasteful behavior. It should be your principal strategy when considering what to do with in-process development.

Closing

In this chapter I discussed 11 important strategies for portfolio planning (portfolio management). My intent was not to provide a cafeteria where you selectively pick and choose which strategies to use. All 11 strategies reinforce one another. You will derive the maximum benefit by doing all of them. That being said, if for some reason I were forced to use just one strategy from each category, I would focus on cost of delay, smaller and more frequent releases, WIP limit, and marginal economics.

In the next chapter I will discuss product planning (envisioning). The output of that process provides us with candidate products to consider during portfolio planning.